EP1000 PLC Quick User Manual

(Version: V1.00)

ESTUN AUTOMATION CO., LTD

Total Solution Supplier

Version History

No.	Date	Version	Description
1	2015-03-20	V1.00	Initial release.

Contents

Chapter 1 Product Overview 1
1.1 Overview 1
1.2 Module Diagram 1
1.3 EPX-Programmer 3
Chapter 2 Installation 4
2.1 Matters Need Attention 4
2.2 Assemblage 5
2.3 Wiring 7
2.3.1 Wiring of CPU Module 7
2.3.2 Wiring of Communication Module 8
2.3.3 Wiring of Input Module 9
2.3.4 Wiring of Output Module 11
2.3.5 Wiring of Analog Module 16
2.3.6 Wiring of Power Supply Module 17
Chapter 3 Basic Operation 18
3.1 Connect to PC 18
3.2 Install and Use EPX-Programmer 18
3.2.1 Installing EPX-Programmer 18
3.2.2 Starting EPX-Programmer 19
3.3 Create a Sample Program 20
3.3.1 Description 20
3.3.2 Procedure 21
3.4 Download the Sample Program 22
3.5 Run the Program 23
3.6 Status Chart 23
Appendix A Modules Dimension 25

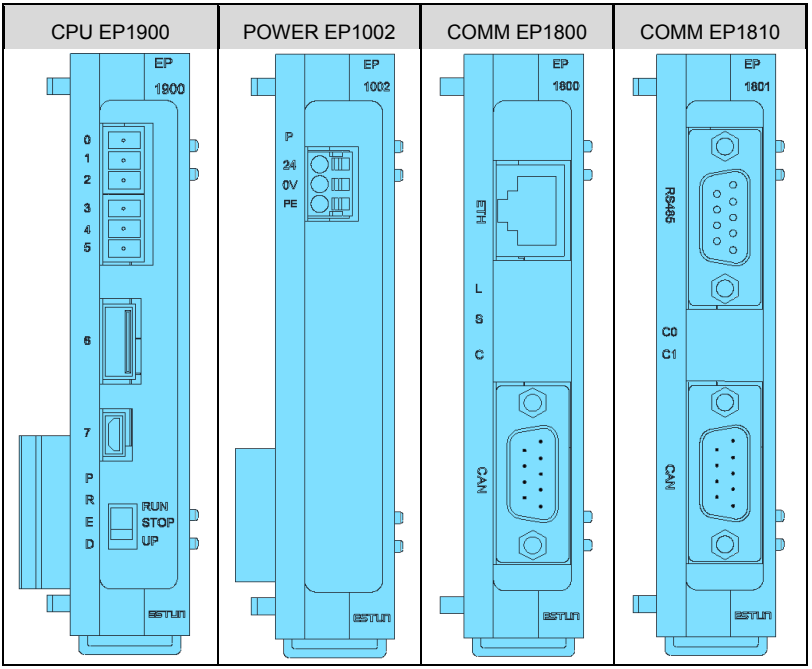
Chapter 1 Product Overview

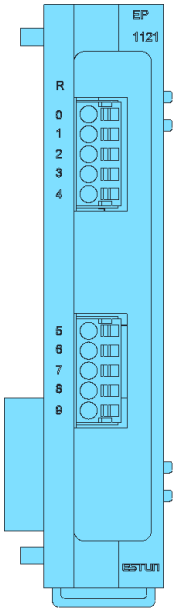
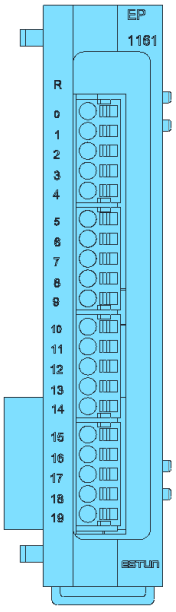
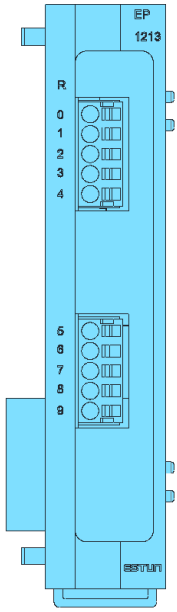
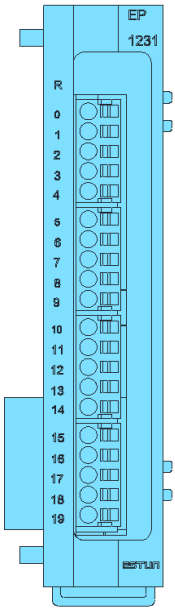
1.1 Overview

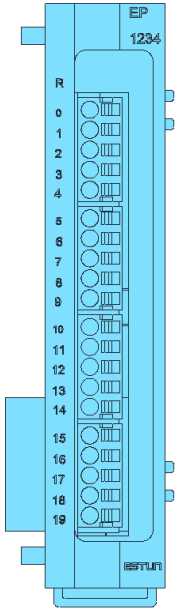
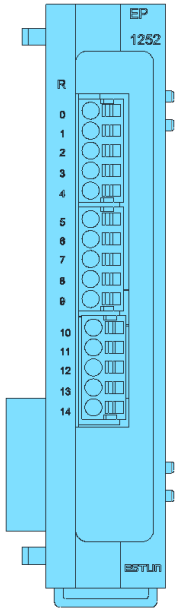
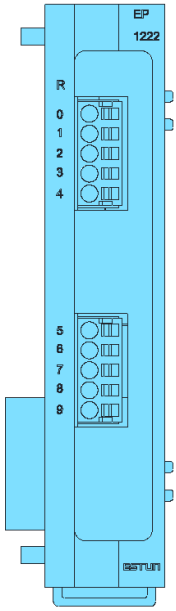
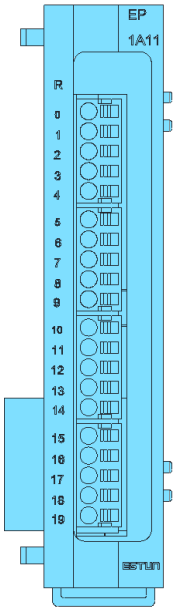
The EP1000 series is a line of networking micro-programmable logic controllers (Networking Micro PLCs) that can control a variety of automation applications. The user program can be composed many instructions, including Bit Logic, Counter, Timer, Math, complicated data processing, and communication, which make the PLC can be monitored the input status and changed the output status for the controlling.

Modular design and a powerful instruction set make the EP1000 a perfect solution for controlling small applications, especially the position and speed controlling system based on CANopen, and the data acquisition system based on Ethernet. The wide variety of EP1000 models and the Windows-based programming tool give you the flexibility you need to solve your automation problems.

1.2 Module Diagram



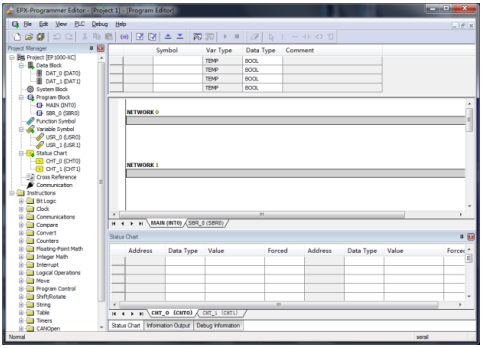
INPUT EP1121	INPUT EP1161	RELAY EP1213	MOS EP1231
			

OUTPUT EP1234	OUTPUT EP1252	OUTPUT EP1222	ANALOG EP1A11
			

1.3 EPX-Programmer

EPX-Programmer Editor makes it easy for you to study and design a micro PLC program, and you can carry out the following task:

- Set up PLC (CPU) type
- Create a PLC program
- Set up communication
- Set up System Block
- Design the program by using LAD or STL
- Set up Variable Symbol
- View the Function Symbol



- Compile, Download, Run and Monitor the program

Chapter 2 Installation

2.1 Matters Need Attention

Installation mode

You can install the EP1000 PLCs either on a panel or on a standard rail, and you can orient the EP100 PLCs either horizontally or vertically.

Warning!

The EP1000 PLCs are Open Type Controllers. It is required that you install the EP1000 in a housing, cabinet, or electric control room. Entry to the housing, cabinet, or electric control room should be limited to authorized personnel.
Failure to follow these installation requirements could result in death or serious injury to personnel, and/or damage to equipment.

Away from interferences

As a general rule for laying out the devices of your system, always separates the devices that generate high voltage and high electrical noise from the low-voltage, logic-type devices such as the EP1000 PLCs, and keep them a minimum distance of 400mm, as shown in Figure 2-1.

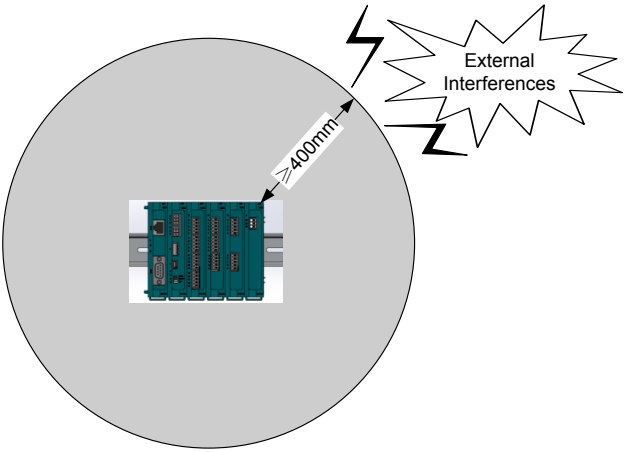


Figure 2-1 Away from the external interferences

Installation space

EP1000 PLCs are designed for natural convection cooling. For proper cooling, you must provide a clearance of at least 150 mm above and below the devices. Also, allow at least 150 mm of depth.

When planning your layout for the EP1000 system, allow enough clearance for the wiring and communications cable connections. For additional flexibility in configuring the layout of the EP1000 system, use the I/O expansion cable.

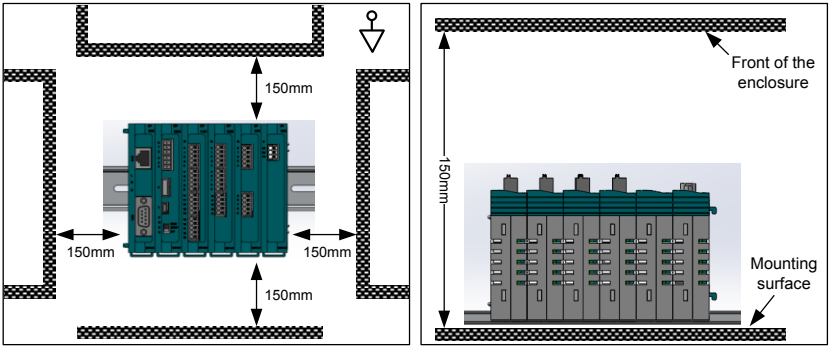
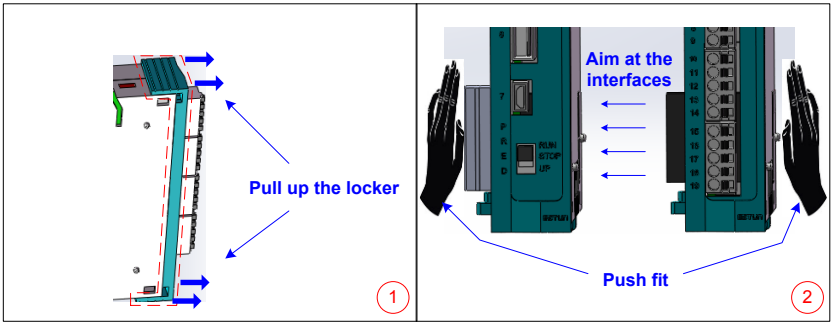


Figure 2-2 Optimal position for the installation

2.2 Assemblage



1 **Pull up the locker of the right module.**

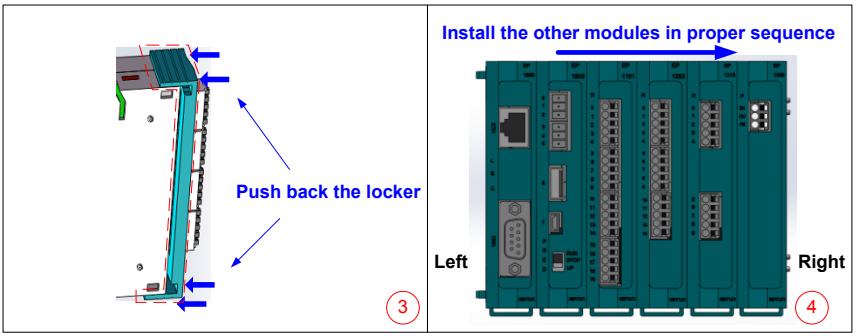
2 **Assemble the modules.**

There are some expansion interfaces at the side of the modules, which can combine with another expansion module. Please aim at the interfaces and push fit.

Caution

There is a toggle switch **S1** on the left of EP1002 module, which is used for identifying the terminal. When the modules have been assembled, the EP1002 module on the very right must turn the toggle switch to **ON**, while the other EP1002 modules must turn the toggle switch to **OFF**.

There are two toggle switches **S1** and **S2** on the left of EP1810 module (EP1800 only has one toggle switch **S1**), which also are used for identifying the terminal respectively. If EP1810 or EP1800 is in the terminal of whole network, the corresponding toggle switch must be turned to **ON**; or else, it must be **OFF**.



3 Push back the locker of the right module.

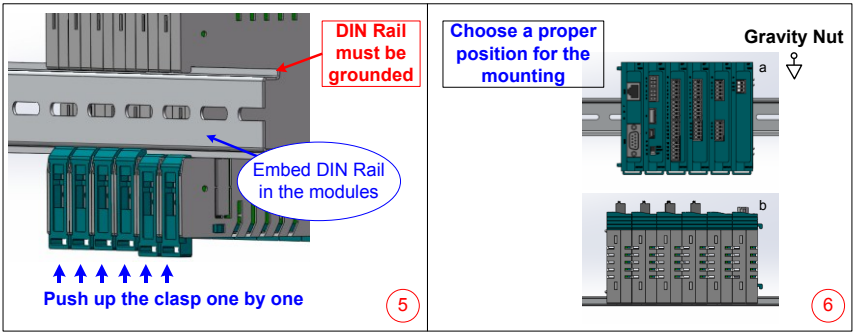
4 Install the other modules in proper sequence.

Caution

The module on the left of CPU module only can be a communication module, please install other modules on the right of CPU module.

If there is no communication module in the PLSs, you must add a head shield on the left interface of CPU module.

The layout of IO modules can be free. However, we suggest you install the same kind modules together. And, the very right of PLCs must be an EP1002 module.



5 Install the modules on the DIN Rail.

6 Choose a proper position for the mounting.

Caution

Make sure the clasp is jointed the DIN Rail correctly, and the DIN Rail must be grounded.

2.3 Wiring

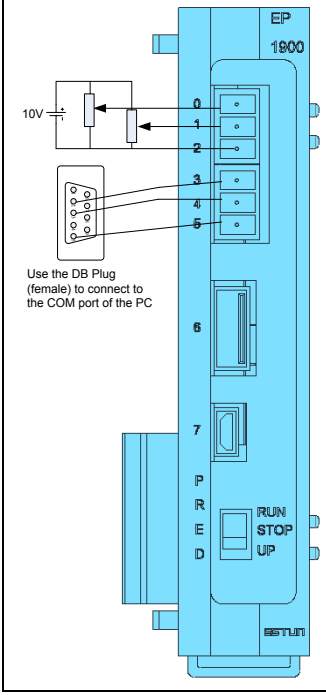
Warning!

Make sure the power supply is cut off before your wiring.

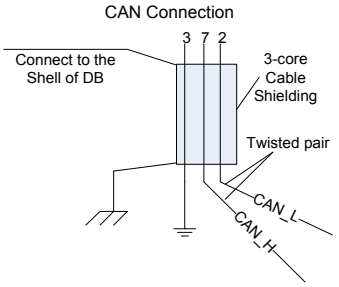
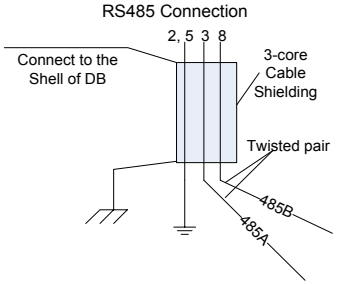
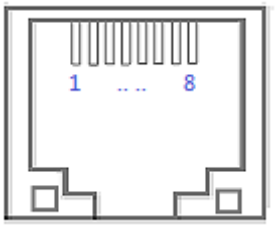
Always follow appropriate safety precautions; Attempts to install or wire the EP1000 PLCs or related equipment with power applied could cause electric shock or faulty operation of equipment.

Since the type of the terminal block is the spring crimp, please use the split conductor for the wiring, which the width is under 1.5 mm², and it is not necessary to add a chuck nut on the stripped wire.

2.3.1 Wiring of CPU Module

Diagram	Pin	Sign	Description
	0	AI1	Two input channels for the analog signal
	1	AI0	
	2	AGND	Ground wire of the analog signal
	3	TX	RS232 ports
	4	RX	
	5	GND	Ground wire of RS232
	6	USB-HOST	Reserved
	7	USB-SLAVE	Reserved

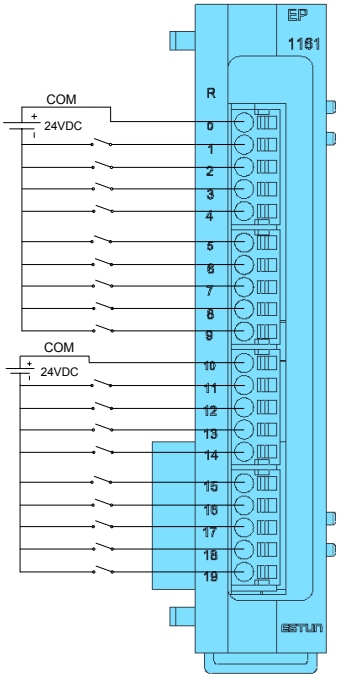
2.3.2 Wiring of Communication Module

Diagram	Pin	Sign	Description
	2	CAN_L	Twisted-pair
	7	CAN_H	
	3	CAN_GND	Ground wire of CAN
	Others	Null	-
	3	485A	Twisted-pair
	8	485B	
	2	GND	Ground wire of RS485
	5	GND	Ground wire of RS485
	Others	Null	-
	1	Tx+ / Rx+	Green and White / Orange and White
	2	Tx- / Rx-	Green / Orange
	3	Rx+ / Tx+	Orange and White / Green and White
	4	-	Blue
	5	-	Blue and White
	6	Rx- / Tx-	Orange / Green
	7	-	Brown and White
	8	-	Brown

Since the Ethernet port of EP1000 has a capability of self-adaption, it is not necessary to consider whether the twisted pair is crossed or directed.

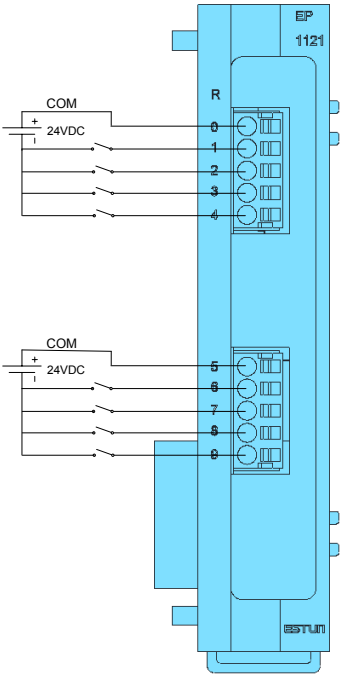
2.3.3 Wiring of Input Module

EP1161

Diagram	Pin	Sign	Description
	0	1M	Common port for the ports of I0 to I8
	1	I0	The input signal ports
	2	I1	
	3	I2	
	4	I3	
	5	I4	
	6	I5	
	7	I6	
	8	I7	
	9	I8	
	10	2M	Common port for the prots of I9 to I17
	11	I9	The input signal ports
	12	I10	
	13	I11	
	14	I12	
	15	I13	
	16	I14	
	17	I15	
	18	I16	
	19	I17	

The common port for the input signals is ambipolar.

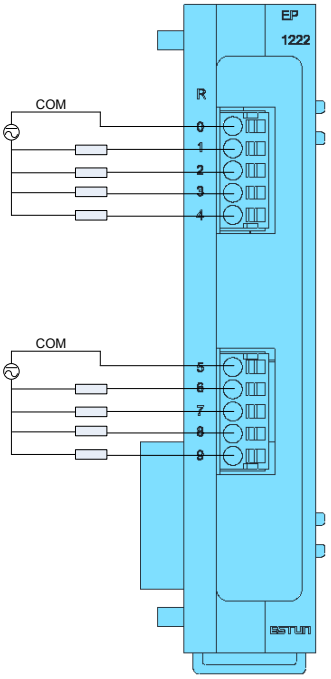
EP1121

<table>
<tr><th>Diagram</th><th>Pin</th><th>Sign</th><th>Description</th></tr>
<tr><td rowspan="10"> 
</td><td>0</td><td>1M</td><td>Common port for the ports of I0 to I3</td></tr>
<tr><td>1</td><td>I0</td><td rowspan="4">The input signal ports</td></tr>
<tr><td>2</td><td>I1</td></tr>
<tr><td>3</td><td>I2</td></tr>
<tr><td>4</td><td>I3</td></tr>
<tr><td>5</td><td>2M</td><td>Common port for the ports of I4 to I7</td></tr>
<tr><td>6</td><td>I4</td><td rowspan="4">The input signal ports</td></tr>
<tr><td>7</td><td>I5</td></tr>
<tr><td>8</td><td>I6</td></tr>
<tr><td>9</td><td>I7</td></tr>
</table>

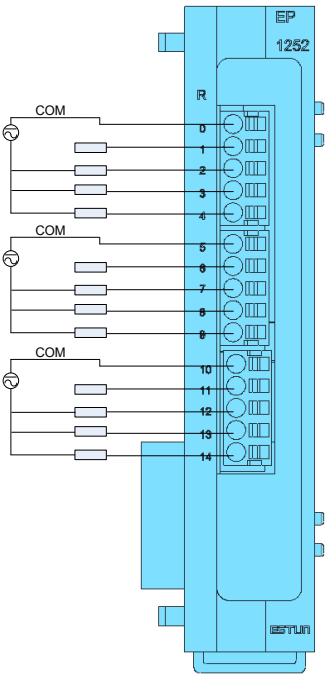
The common port for the input signals is ambipolar.

2.3.4 Wiring of Output Module

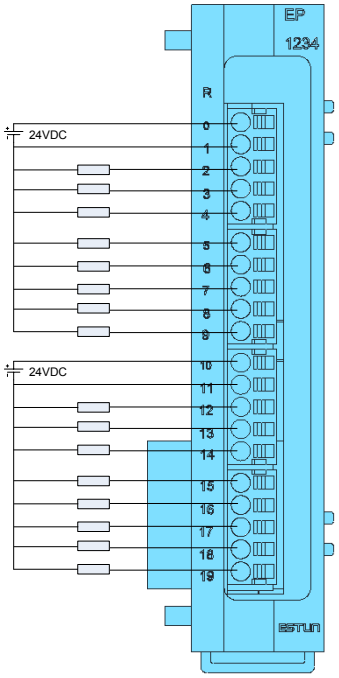
EP1222

<table>
<tr><th>Diagram</th><th>Pin</th><th>Sign</th><th>Description</th></tr>
<tr><td rowspan="10"> 
</td><td>0</td><td>1L</td><td>Common port for the ports of Q0 to Q3</td></tr>
<tr><td>1</td><td>Q0</td><td rowspan="4">The output signal ports</td></tr>
<tr><td>2</td><td>Q1</td></tr>
<tr><td>3</td><td>Q2</td></tr>
<tr><td>4</td><td>Q3</td></tr>
<tr><td>5</td><td>2L</td><td>Common port for the ports of Q4 to Q7</td></tr>
<tr><td>6</td><td>Q4</td><td rowspan="4">The output signal ports</td></tr>
<tr><td>7</td><td>Q5</td></tr>
<tr><td>8</td><td>Q6</td></tr>
<tr><td>9</td><td>Q7</td></tr>
</table>

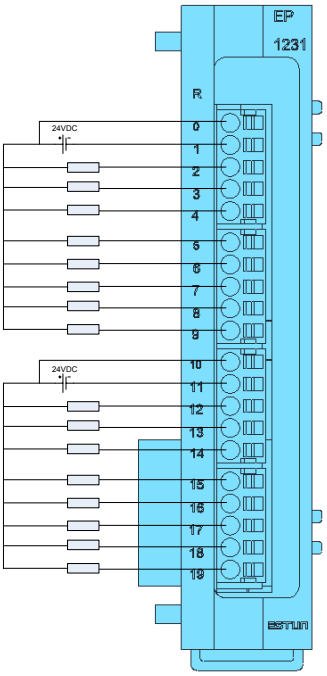
EP1252

Diagram	Pin	Sign	Description
	0	1L	Common port for the ports of Q0 to Q3
	1	Q0	The output signal ports
	2	Q1	
	3	Q2	
	4	Q3	
	5	2L	Common port for the ports of Q4 to Q7
	6	Q4	The output signal ports
	7	Q5	
	8	Q6	
	9	Q7	
	10	3L	Common port for the ports of Q8 to Q11
	11	Q8	The output signal ports
	12	Q9	
	13	Q10	
	14	Q11	

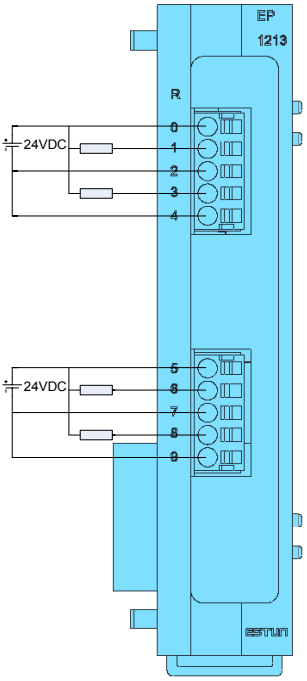
EP1234

Diagram	Pin	Sign	Description
	0	0V	+24V
	1	1L	0V
	2	Q0	The output signal ports
	3	Q1	
	4	Q2	
	5	Q3	
	6	Q4	
	7	Q5	
	8	Q6	
	9	Q7	
	10	0V	+24V
	11	2L	0V
	12	Q8	The output signal ports
	13	Q9	
	14	Q10	
	15	Q11	
	16	Q12	
	17	Q13	
	18	Q14	
	19	Q15	

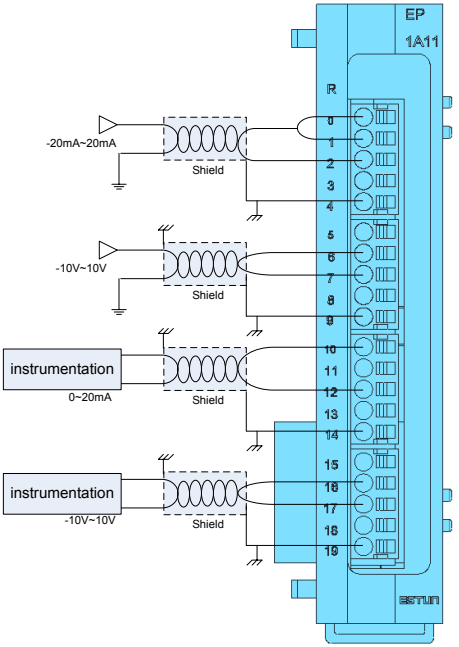
EP1231

Diagram	Pin	Sign	Description
	0	0V	+24V
	1	1L	0V
	2	Q0	The output signal ports
	3	Q1	
	4	Q2	
	5	Q3	
	6	Q4	
	7	Q5	
	8	Q6	
	9	Q7	
	10	0V	+24V
	11	2L	0V
	12	Q8	The output signal ports
	13	Q9	
	14	Q10	
	15	Q11	
	16	Q12	
	17	Q13	
	18	Q14	
	19	Q15	

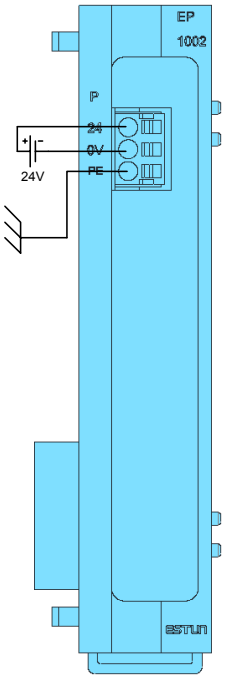
EP1213

Diagram	Pin	Sign	Description
	0	1L	+24V
	1	Q0	The output signal port
	2	Q0-	0V
	3	Q1	The output signal port
	4	Q1-	0V
	5	2L	+24V
	6	Q2+	The output signal port
	7	Q2-	0V
	8	Q3+	The output signal port
	9	Q3-	0V

2.3.5 Wiring of Analog Module

<table>
<tr><th>Diagram</th><th>Pin</th><th>Sign</th><th>Description</th></tr>
<tr><td rowspan="19"> 
</td><td>0</td><td>R0</td><td rowspan="3">Short R0 and 0+, then combine with 0-: Channel 0 is used for current sampling; Combine 0+ with 0-: Channel 0 is used for voltage sampling.</td></tr>
<tr><td>1</td><td>0+</td></tr>
<tr><td>2</td><td>0-</td></tr>
<tr><td>3</td><td>NC</td><td>Reserved</td></tr>
<tr><td>4</td><td>FG</td><td>Ground Terminal</td></tr>
<tr><td>5</td><td>R1</td><td rowspan="3">Short R1 and 1+, then combine with 1-: Channel 0 is used for current sampling; Combine 1+ with 1-: Channel 1 is used for voltage sampling.</td></tr>
<tr><td>6</td><td>1+</td></tr>
<tr><td>7</td><td>1-</td></tr>
<tr><td>8</td><td>NC</td><td>Reserved</td></tr>
<tr><td>9</td><td>FG</td><td>Ground Terminal</td></tr>
<tr><td>10</td><td>I0</td><td rowspan="3">Combine I0 with M0: Channel 0 is used for current inputting; Combine V0 with M0: Channel 0 is used for Voltage inputting.</td></tr>
<tr><td>11</td><td>V0</td></tr>
<tr><td>12</td><td>M0</td></tr>
<tr><td>13</td><td>NC</td><td>Reserved</td></tr>
<tr><td>14</td><td>FG</td><td>Ground Terminal</td></tr>
<tr><td>15</td><td>I1</td><td rowspan="3">Combine I1 with M1: Channel 1 is used for current inputting; Combine V1 with M1: Channel 0 is used for Voltage inputting.</td></tr>
<tr><td>16</td><td>V1</td></tr>
<tr><td>17</td><td>M1</td></tr>
<tr><td>18</td><td>NC</td><td>Reserved</td></tr>
<tr><td>19</td><td>FG</td><td>Ground Terminal</td></tr>
</table>

2.3.6 Wiring of Power Supply Module

Diagram	Pin	Sign
	1	+24V
	2	0V
	3	PE

Chapter 3 Basic Operation

3.1 Connect to PC

To carry out your programming, you need connect the PLC to a PC. We suggest you using the standard cable of RS232 for the connecting, as shown in Figure 3-1.

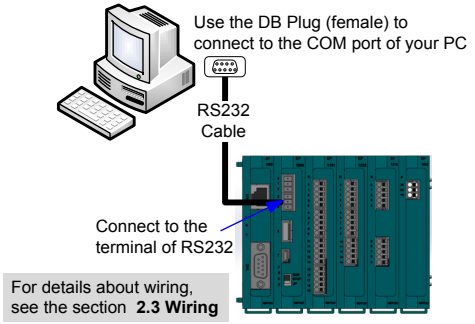


Figure 3-1 Connect to a PC

If the EP1800 module is in your PLCs, you can also use a standard Ethernet **cross twisted-pair** for connecting the Ethernet interface of the EP1800 module.

3.2 Install and Use EPX-Programmer

3.2.1 Installing EPX-Programmer

Computer requirements

EPX-Programmer editor runs on a personal computer, and your computer should meet the following minimum requirements:

- Operating System: Windows XP
- ROM: At least 150M bytes of free hard disk space
- Auxiliary Device: Mouse (recommended), Keyboard

Installation steps

1 Get **EPX-Programmer** installation package.
2 Copy the installation package to your PC, and double-click to run it.
3 Follow the information of the page to install EPX-Programmer.
----END

3.2.2 Starting EPX-Programmer

1 Open the EPX-Programmer editor.

When success in installing, there is a shortcut icon **EPX-Programmer.exe** created on your desk. Double-click icon to open a new project, as shown in Figure 3-2.

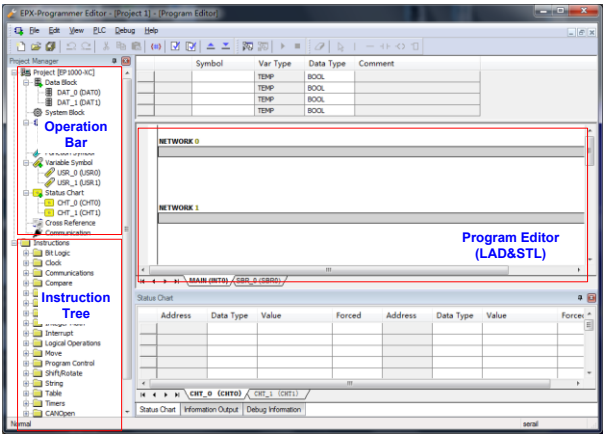


Figure 3-2 New EPX-Programmer Project

2 Verifying the communication parameters for EPX-Programmer.

Click on the **Communication** icon in **Operation Bar**, and use the **Communication** dialog box to connect with your PLC, as shown in **错误!未找到引用源。**.

You need set the parameters on the dialog box, according to your actual using port.

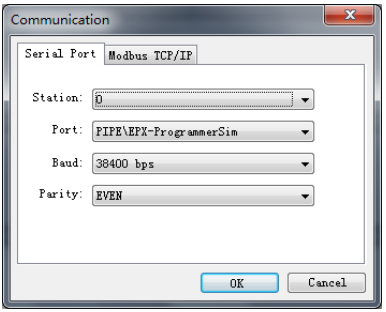


Figure 3-3 Communication Dialog Box

3 Establishing Communications with the PLC.

After the communication setting is done, click the **Debug > Connect** menu.

- If the communication is normal, the status bar will show **Online**.
- If the communication is abnormal, it will pop up an Error dialog box.

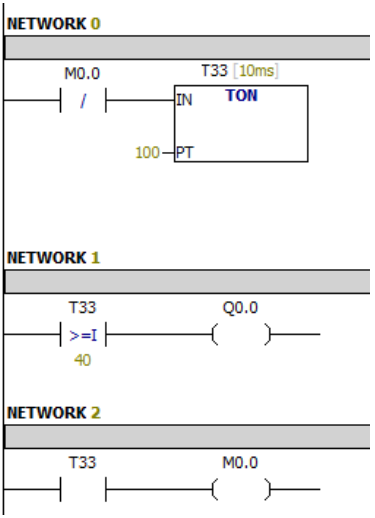
----END

3.3 Create a Sample Program

3.3.1 Description

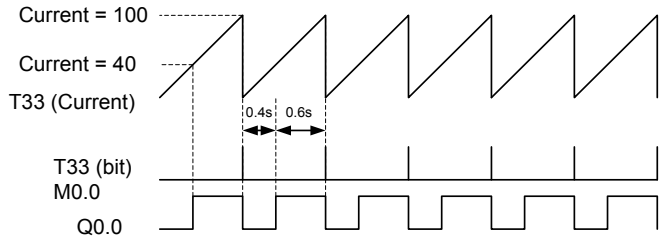
Entering this example of a control program will help you understand how easy it is to use EPX-Programmer. This program uses six instructions in three networks to create a very simple, self-starting timer that resets itself.

For this example, you use the Ladder (LAD) editor to enter the instructions for the program. The following example shows the complete program in both LAD and Statement List (STL). The network comments in the STL program explain the logic for each network. The timing diagram shows the operation of the program.




```
NETWORK 0
10 ms timer T33 times out after (100 x 10 ms = 1 s).
M0.0 pulse is too fast to monitor with Status view.
LDN M0.0
TON T33 [10ms], 100
NETWORK 1
Comparison becomes true at a rate that is
visible with Status view. Turn on Q0.0 after
(40 x 10 ms = 0.4 s),
for a 40% OFF/60% ON waveform.
LDW>= T33, 40
= Q0.0
NETWORK 2
T33 (bit) pulse too fast to monitor with Status view.
Reset the timer through M0.0 after the
(100 x 10 ms = 1 s) period.
LD T33
= M0.0
```

Timing Diagram



3.3.2 Procedure

Step 1 Open EPX-Programmer editor.

Step 2 Entering Network 1: Starting the Timer.

When M0.0 is off (0), this contact turns on and provides power flow to start the timer. To enter the contact for M0.0:

1. Either double-click the Bit Logic icon or click on the plus sign ⊞ to display the bit logic instructions.
2. Select the Normally Closed contact.
3. Hold down the left mouse button and drag the contact onto the first network.
4. Click on the **???** above the contact and enter the following address: **M0.0**
5. Press the Return key to enter the address for the contact.

To enter the timer instruction for T33:

1. Double-click the Timers icon to display the timer instructions.
2. Select the TON (On-Delay Timer).
3. Hold down the left mouse button and drag the timer onto the first network.
4. Click on the **???** above the timer box and enter the following timer number: **T33**
5. Press the Return key to enter the timer number and to move the focus to the preset time (PT) parameter.
6. Enter the following value for the preset time: **100**
7. Press the Return key to enter the value.

Step 3 Entering Network 2: Turning the Output On.

When the timer value for T33 is greater than or equal to 40 (40 times 10 milliseconds, or 0.4seconds), the contact provides power flow to turn on output Q0.0 of the EP1000. To enter the Compare instruction:

1. Double-click the Compare icon to display the compare instructions. Select the >=I instruction (Greater-Than-Or-Equal-To-Integer).
2. Hold down the left mouse button and drag the compare instruction onto the second network.
3. Click on the **???** above the contact and enter the address for the timer value: **T33**
4. Press the Return key to enter the timer number and to move the focus to the other value to be compared with the timer value.
5. Enter the following value to be compared with the timer value: **40**
6. Press the Return key to enter the value.

To enter the instruction for turning on output Q0.0:

1. Double-click the Bit Logic icon to display the bit logic instructions and select the output coil.

2 Hold down the left mouse button and drag the coil onto the second network.
3 Click on the **???** above the coil and enter the following address: **Q0.0**.
4 Press the Return key to enter the address for the coil.

Step 4 Entering Network 3: Resetting the Timer.

When the timer reaches the preset value (100) and turns the timer bit on, the contact for T33 turns on. Power flow from this contact turns on the M0.0 memory location. Because the timer is enabled by a Normally Closed contact for M0.0, changing the state of M0.0 from off (0) to on (1) resets the timer. To enter the contact for the timer bit of T33:

1 Select the Normally Open contact from the bit logic instructions.
2 Hold down the left mouse button and drag the contact onto the third network.
3 Click on the **???** above the contact and enter the address of the timer bit: **T33**
4 Press the Return key to enter the address for the contact.

To enter the coil for turning on M0.0:

1 Select the output coil from the bit logic instructions.
2 Hold down the left mouse button and drag the output coil onto the third network.
3 Double-click the **???** above the coil and enter the following address: **M0.0**
4 Press the Return key to enter the address for the coil.

Step 5 Saving the Sample Project.

After entering the three networks of instructions, you have finished entering the program. When you save the program, you create a project that includes the S7-200 CPU type and other parameters. To save the project:

1 Select the **File > Save As** menu command from the menu bar.
2 Enter a name for the project in the **Save As** dialog box.
3 Click **OK** to save the project.

After saving the project, you can download the program to the EP1000 PLC.

----END

3.4 Download the Sample Program

1 Open the sample program.
2 Select the **File > Download** menu command from the menu bar.
3 Click **OK** on the pop-up dialog box to download the elements of the program to the EP1000 PLC.

3.5 Run the Program

When success in downloading, follow the above steps to run the program:

1 Select the **PLC > RUN** menu command from the menu bar.
2 Click **OK** on the pop-up dialog box.

When the EP1000 PLC goes to RUN mode, the output LED for Q0.0 turns on and off as the EP1000 PLC executes the program.

Congratulations! You have just completed your first EP1000 program.

EPX-Programmer displays the values for the instructions. To stop the program, place the EP1000 PLC in **STOP** mode by clicking the STOP icon or by selecting the **PLC > STOP** menu command.

3.6 Status Chart

A status chart allows you to monitor or modify the values of the process variables as your PLC runs the control program. You can track the status of program inputs, outputs, or variables by displaying the current values. The status chart also allows you to force or change the values of the process variables.

You can create multiple status charts in order to view elements from different portions of your program. In general, a **Status Chart** is automatically displayed below the program editor area. You can check or not check the **View > Frame > Status Chart** menu command to turns on or off the **Status Chart**.

When you create a status chart, you enter addresses of process variables for monitoring. You cannot view the status of constants, accumulators, or local variables. You can display a timer or counter value either as a bit or as a word. Displaying the value as a bit shows the status of the timer or counter bit; displaying the value as a word shows the timer or counter value.

You can use the Status Chart to force values. As shown in **错误!未找到引用源。**, to force a new value, select **Forced** option in the Forced column of the Status Chart, then enter the desired value in the Value column.

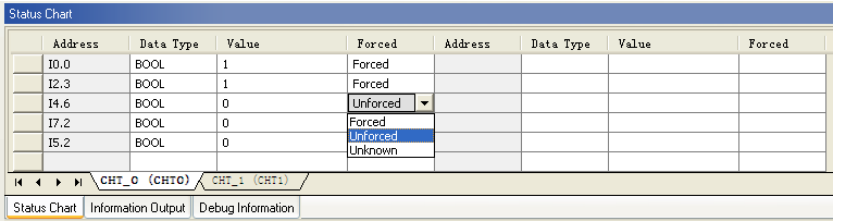


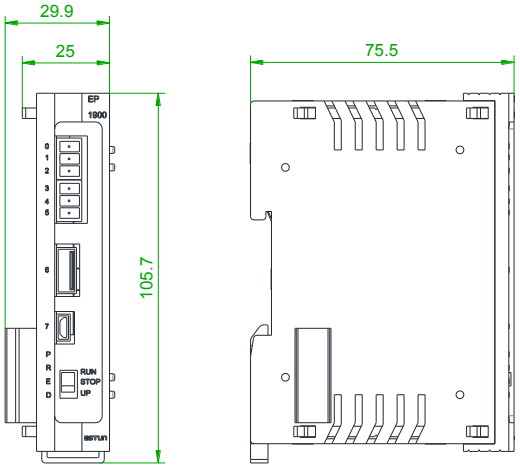
Figure 3-4 Force a value

The EP1000 PLC only allows you to force any or all of the I/O points (I and Q bits).

Appendix A Modules Dimension

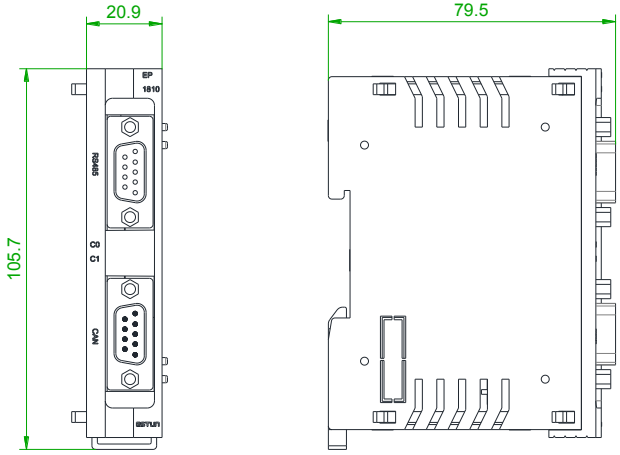
CPU EP1900

Unit: millimeter



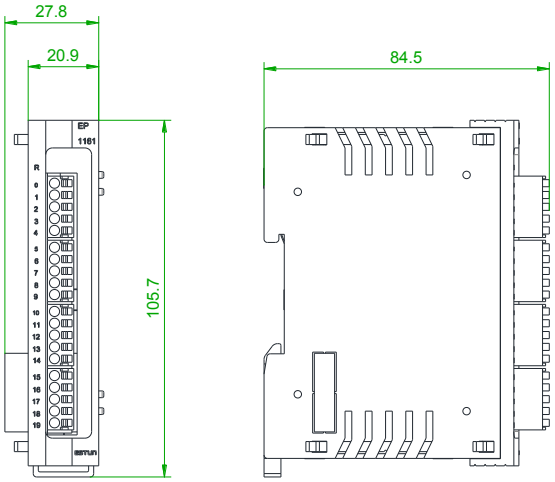
COMM EP18x0

Unit: millimeter



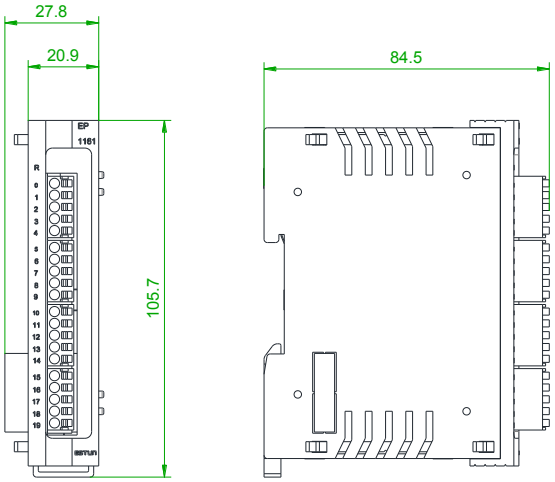
INPUT EP1161

Unit: millimeter



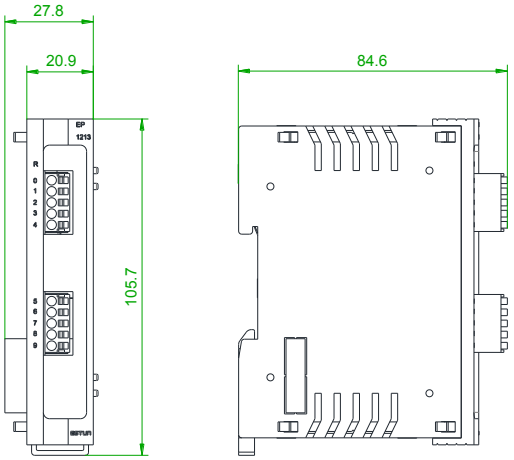
RELAY EP1252

Unit: millimeter



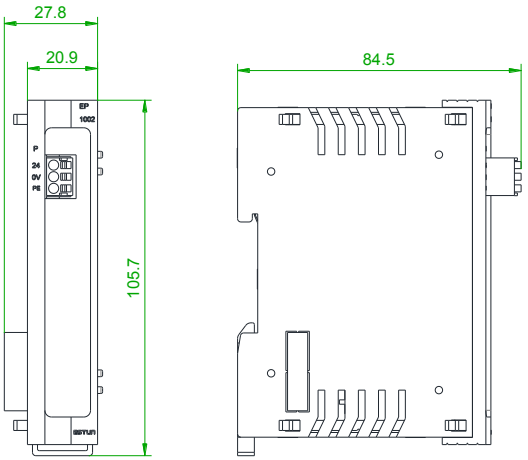
MOS EP1213

Unit: millimeter



POWER EP1002

Unit: millimeter



ESTUN AUTOMATION CO., LTD

Add: 155 Jiangjun Road, Jiangning Development Zone,
Nanjing 211106, P.R.China
TEL: 025-52785866
FAX: 025-52785992
WEB: www.estun.com
Email: info@estun.com

www.estun.com

ESTUN